



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/849,099	05/04/2001	Praerit Garg	MSFT-0223/158385.1	7971
7590	06/14/2005		EXAMINER	
WOODCOCK WASHBURN KURTZ MACKIEWICZ & NORRIS LLP One Liberty Place -46th Floor Philadelphia, PA 19103			LEMMAN, SAMSON B	
			ART UNIT	PAPER NUMBER
			2132	

DATE MAILED: 06/14/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	Application No.	Applicant(s)	
	09/849,099	GARG ET AL.	
	Examiner Samson B. Lemma	Art Unit 2132	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) Responsive to communication(s) filed on 21 March 2005.
- 2a) This action is FINAL.                    2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) Claim(s) 1-21 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) Claim(s) \_\_\_\_\_ is/are allowed.
- 6) Claim(s) 1-21 is/are rejected.
- 7) Claim(s) \_\_\_\_\_ is/are objected to.
- 8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on \_\_\_\_\_ is/are: a) accepted or b) objected to by the Examiner.  
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) All    b) Some \* c) None of:
1. Certified copies of the priority documents have been received.
  2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) Notice of References Cited (PTO-892)  
 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)  
 3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
 Paper No(s)/Mail Date \_\_\_\_\_
- 4) Interview Summary (PTO-413)  
 Paper No(s)/Mail Date. \_\_\_\_\_ .
- 5) Notice of Informal Patent Application (PTO-152)
- 6) Other: \_\_\_\_\_.

## **DETAILED ACTION**

1. This office action is in reply to an amendment filed on **March 21, 2005**.  
**Claims 1-21** are pending.
2. With respect to the objection made to the specification in relation to the term "ACE" applicant reconfirms the examiner interpretation and since the term is already defined on page 3, line 1, as applicant points out, the objection is withdrawn.
3. Acknowledgement is made to the correction made by the applicant to overcome the 35 U.S.C. 112, second paragraph for claim 1.

## ***Response to Arguments***

4. Applicant's argument filed on **March 21, 2005** have been fully considered but they are not persuasive.

The first argument by the applicant is about the independent claims 1, 14 and 18 includes limitations that are not shown or suggested by the combinations of the references on the record, namely **Netegrity White Paper**.

**Applicant argued** that the merely caching policy information pursuant to accessing to a resource does not amount to determining a static maximum allowed data structure in accordance with the invention. Applicants argued that the term "dynamically" in the Netegrity While paper refers only to how the caches are filled over time as access request are processed (as opposed to initialization of the

Art Unit: 2132

system.) and this does not in any way relate to the difference between static and dynamic access policy addressed by the invention. Applicants secondly further argued that the office action improperly focuses the 103 obviousness analysis on the teachings of applicants specification when the claim language should be the focus of the analysis. For instance, even if it was true that the Netegrity white paper also relates to reducing the burden of redundant access checks, this fact would be irrelevant to the claim analysis.

**Examiner disagrees with the above argument.**

Examiner would point out that the term “dynamically” as used in the claim is much more closely relates to the Netegrity While paper referring since it is related as applicant points out how the caches are filled over time as access request are processed. Examiner further points out the that Netegrity While paper further discloses the following, “When the web agent is initialized, it establishes or enforces a static and dynamic access policy or cache of information protecting a resource by the web agent as explained on page 2, reference “Resource Cache “ and page3, Paragraph 1-5, and page 2, last Paragraph”.

Furthermore, the recitation “A method of enforcing static and dynamic access policy...” has not been given patentable weight because the recitation occurs in the preamble. A preamble is generally not accorded any patentable weight where it merely recites the purpose of a process or the intended use of a structure, and where the body of the claim does not depend on the preamble for completeness but, instead, the process steps or structural limitations are able to stand alone. See *In re Hirao*, 535 F.2d 67, 190 USPQ 15 (CCPA 1976) and *Kropa v. Robie*, 187 F.2d 150, 152, 88 USPQ 478, 481 (CCPA 1951).

**As to the 2<sup>nd</sup> argument made by the applicant,** the Examiner points out that, the office action did not improperly focuses on the applicant specification instead interprets

Art Unit: 2132

the claim language in the eye's of the applicant specification. This interpretation is actually relevant and proper as it clarifies the examiner interpretation of the claim language. The term used by the applicant in the argued claims for instance "static maximum allowed access data structure" is not a common term for one of ordinary skill in the art and has to be interpreted in the eye's of the applicant specification and this is what is done by the examiner.

In order to show how the term used in the claim limitation is related to the reference used by the examiner, the examiner points out the following.

"**Netegrity** discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system, (Page 2, reference "Resource Cache" , under the title web Agent Caches" and "page 3, Paragraph 1-5") (When the web agent is initialized, it establishes or enforces a static and dynamic access policy or cache of information protecting a resource by the web agent as explained on page 2, reference "Resource Cache " and page3, Paragraph 1-5, and page 2, last Paragraph)

- The system having a client thereof making a first access request for the resource, the method comprising: (Page 2, reference "Resource Cache" and "User session Cache")
- Determining a static maximum allowed access data structure pursuant to an evaluation of the first access request, wherein the static maximum allowed access data structure includes information representative of a set of policies that is reduced to static form that is common to a class of access requests; (Page 2, and Page 3. Paragraph 1-5)  
("Applicant defined on the 1<sup>st</sup> page of the disclosure that the invention is about re-using the computations that have already been made, so that policy evaluations are

Art Unit: 2132

not repeated, thereby making a system more efficient, freeing up computer resources and generally increasing performance. Applicant on page 3, 2<sup>nd</sup> and 3<sup>rd</sup> paragraphs, explained how several access checks involves the same user accessing resources protected by the same authorization policy and caching this particular access policy determination that is likely to be repeated called by the applicant as “static maximum allowed access” and that is granted for given access inquiry and ultimately cached. **Netegrity** on page 2, 2<sup>nd</sup> paragraph, under the title “web agent caches” discloses that the web agent has two caches to optimize performance by saving the information that is likely to be repeated on either resource or sessions cache or both. This information which is saved is interpreted by the office as “static maximum allowed access” )

- Storing the static maximum allowed access data structure; (Page 2, 2<sup>nd</sup> paragraph, under the title “web agent caches”) and
- In response to a determination that the static maximum allowed access data structure is applicable to a second access request, utilizing said static maximum allowed access data structure in connection with the requested permission set of the second access request. (Page 2, 2<sup>nd</sup> paragraph, under the title “web agent caches”) (When any subsequent access or second access request is attempted/made for the resource, the web agent will determine whether the already stored “static maximum allowed access data structure” is applicable for the second or subsequent request by looking into the local memory which has already stored the information which is interpreted by the office as “the maximum allowed access data” without having to go the policy server. This optimizes performance. And on the side, after the user is authenticated, the web agent also caches the information about the user which

allows second access request or subsequent operations to utilize the already stored information which is interpreted by the office as “the maximum allowed access data” either to this resources or to other resources protected by the same policies to be resulting in great optimization”

**The third argument by the applicant** is about the independent claims 14, which applicant argued that it includes limitation that the static maximum allowed access mechanism provides extensible support for application-defined business rules via a set of APIs and DACLs. The Netegrity Paper includes no such teaching or suggestion.

**Examiner disagrees with the above argument.**

**In response to the applicant’s argument the Examiner points out the argument** raised by the applicant is similar to the one described above. Examiner response given above is also applicable to this argument.

**The next argument by the applicant is about claim 18, which applicant argued that it requires** static maximum allowed access data structure including an identifier that is not mentioned or suggested by the reference/s on the record.

**Examiner disagrees with the above argument.**

**In response to the applicant’s argument the Examiner points out the argument** raised by the applicant is similar to the one described for claim 1 above. Examiner response given above and in the office action is also applicable to this argument.

**Applicant’s last argument is regarding the dependent claims.**

Applicants argued that since the independent claims are patentable therefore all the claims dependent thereon are also in condition for allowance for the same reasons argued for the independent claims.

**In response to the above argument by the applicant, the examiner replay discussed** for the independent claims above is also valid towards this argument.

Art Unit: 2132

Therefore every elements of the limitation of the claims including the newly added limitation to some of the claims is explicitly, implicitly or inherently suggested and disclosed by the combinations of the references on the record and the rejection remains valid.

### ***Claim Rejections - 35 USC § 103***

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. **Claims 1-13** are rejected under 35 U.S.C. 103(a) as being unpatentable over the provided Information disclosure by the applicant in particular **Netegrity white paper**, “SiteMinder Delivers Industry-Leading Performance, Scalability, and Reliability (hereinafter referred to as **Netegrity**) (December 1999) in view of **Schneck et al**, (hereinafter referred as **Schneck**)(U.S. Publication Number: 2001/0021926A1)

7. **As per claims 1, 12-14,** **Netegrity** discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system, (Page 2, reference “Resource Cache” , under the title web Agent Caches” and “page 3, Paragraph 1-5”) (When the web agent is initialized, it establishes or enforces a static and dynamic access policy or cache of information protecting a resource by the web agent as explained on page 2, reference “Resource Cache “ and page3, Paragraph 1-5, and page 2, last Paragraph)

- The system having a client thereof making a first access request for the resource, the method comprising: (Page 2, reference “Resource Cache” and “User session Cache”)
- Determining a static maximum allowed access data structure pursuant to an evaluation of the first access request, wherein the static maximum allowed access data structure includes information representative of a set of policies that is reduced to static form that is common to a class of access requests; (Page 2, and Page 3. Paragraph 1-5)

(“Applicant defined on the 1<sup>st</sup> page of the disclosure that the invention is about reusing the computations that have already been made, so that policy evaluations are not repeated, thereby making a system more efficient, freeing up computer resources and generally increasing performance. Applicant on page 3, 2<sup>nd</sup> and 3<sup>rd</sup> paragraphs, explained how several access checks involves the same user accessing resources protected by the same authorization policy and caching this particular access policy determination that is likely to be repeated called by the applicant as “static maximum allowed access” and that is granted for given access inquiry and ultimately cached. **Netegrity** on page 2, 2<sup>nd</sup> paragraph, under the title “web agent caches” discloses that the web agent has two caches to optimize performance by saving the information that is likely to be repeated on either resource or sessions cache or both. This information which is saved is interpreted by the office as “static maximum allowed access” )
- Storing the static maximum allowed access data structure; (Page 2, 2<sup>nd</sup> paragraph, under the title “web agent caches”) and

Art Unit: 2132

- In response to a determination that the static maximum allowed access data structure is applicable to a second access request, utilizing said static maximum allowed access data structure in connection with the requested permission set of the second access request. (Page 2, 2<sup>nd</sup> paragraph, under the title “web agent caches”) (When any subsequent access or second access request is attempted/made for the resource, the web agent will determine whether the already stored “static maximum allowed access data structure” is applicable for the second or subsequent request by looking into the local memory which has already stored the information which is interpreted by the office as “the maximum allowed access data” without having to go the policy server. This optimizes performance. And on the side, after the user is authenticated, the web agent also caches the information about the user which allows second access request or subsequent operations to utilize the already stored information which is interpreted by the office as “the maximum allowed access data” either to this resources or to other resources protected by the same policies to be resulting in great optimization)

**Netegrity** does not explicitly teach the how “the static maximum allowed access data” is determined.

However, in the same field of endeavor, **Schneck** discloses how the access control quantities can be determined by including some items including an “allowable size of read-access to the data.” (Column 14, reference [0244], and column 21, claim 20)

It would have been obvious to one having ordinary skill in the art, at the time the invention was made, to combine the techniques of determination of an allowable size as per teachings **Schneck** in to the method of as taught by **Netegrity** in order to increase the performance and optimization of the resources.

Art Unit: 2132

8. **As per claims 2 and 15**, the combinations of **Netegrity** and **Schneck** discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system as applied to claim 1 and 14 above. Furthermore, **Netegrity** discloses the method wherein the storing of the static maximum allowed access data structure includes storing the static maximum allowed access data structure in cache memory. (Page 2)

9. **As per claims 3 and 16** the combinations of **Netegrity** and **Schneck** discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system as applied to claim 1 and 14 above. Furthermore, **Netegrity** discloses the method further comprising computing a client security context after the first access request for the resource is received from the client.(Page 2, paragraph 3, Under the title "user session cache ") (Client is authenticated and this meets the recitation of the limitation)

10. **As per claims 4 and 11** the combinations of **Netegrity** and **Schneck** discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system as applied to claim 1 above. Furthermore, **Netegrity** discloses the method further comprising determining whether said second access request is granted based at least in part on dynamic data and dynamic policy algorithms. (Page 3, 3<sup>rd</sup> paragraph, under the title "Authorization Cache, level 2 Policy cache ")

11. **As per claims 5-7 and 17** the combinations of **Netegrity** and **Schneck** discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system as applied to claim 1. Furthermore **Netegrity** discloses the method further comprising: evaluating whether the requested permission set of the second access request is represented within the static maximum allowed access data structure. (Page 2, 2<sup>nd</sup> paragraph, under the title "web agent caches") (When any subsequent access or second access request is attempted/made for the resource, the web

Art Unit: 2132

agent will determine whether the already stored “static maximum allowed access data structure” is applicable for the second or subsequent request by looking into the local memory which has already stored the information which is interpreted by the office as “the maximum allowed access data” without having to go the policy server, this optimizes performance. And on the side, after the user is authenticated, the web agent also caches the information about the user which allows second access request or subsequent operations to utilize the already stored information either to this resources or to other resources protected by the same policies to be greatly optimized and this meets the recitation of this limitation)

12. As per claims 8-9 the combinations of **Netegrity** and **Schneck** discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system as applied to claim 1. Furthermore **Netegrity** discloses the method wherein evaluating whether there is at least one dynamic access control entry in a discretionary access control list associated with the second access request.(Page 2, and Page 3, 3<sup>rd</sup> paragraph, under the title “Authorization Cache (level 2 Policy cache) (DAC or Discretionary access control is used to control access by restricting a subject’s access to an object. The user is evaluated or authorized as explained on Page 3, 3<sup>rd</sup> paragraph, under the title “Authorization Cache level 2 Policy cache” and this meets the recitation of the limitation)

13. As per claims 10 the combinations of **Netegrity** and **Schneck** discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system as applied to claim 1. Furthermore **Netegrity** discloses the method wherein if there is not at least one deny access control entry, the method further comprises: evaluating whether the requested permission set of the second access request is

Art Unit: 2132

encompassed by (1) permissions obtained by evaluating at least one dynamic grant access control entry and (2) permissions contained said static maximum allowed access data structure. (Page 3, 3<sup>rd</sup> paragraph, under the title "Authorization Cache level 2 Policy cache15.")

14. **Claims 18-21** are rejected under 35 U.S.C. 103(a) as being unpatentable over the provided Information disclosure by the applicant in particular **Netegrity white paper**, "SiteMinder Delivers Industry-Leading Performance, Scalability, and Reliability (hereinafter referred to as **Netegrity**) (December 1999) in view of **Clifton**, (hereinafter referred as **Clifton**)(U.S. Patent. No 5,469,556)

15. **As per claims 18,20 and 21** **Netegrity** discloses a static maximum allowed access data structure stored on a computer readable medium for use in connection with access check determinations for an application in a computer system, the data structure comprising:

- An identifier identifying the data structure as a static maximum allowed access data structure; (Page 2, and Page 3. Paragraph 1-5)

(Applicant on page 3, 2<sup>nd</sup> and 3<sup>rd</sup> paragraphs, explained how several access checks involves the same user accessing resources protected by the same authorization policy and caching this particular access policy determination that is likely to be repeated called by the applicant as "static maximum allowed access". This information "static maximum allowed access" is granted for given access inquiry and ultimately saving computer resources . **Netegrity** on page 2, 2<sup>nd</sup> paragraph, under the title "web agent caches" discloses that the web agent has two caches to optimize performance by saving the information that is likely to be repeated on either "resource" or "sessions cache" or both. This information which is saved is

interpreted by the office as "static maximum allowed access" and this information is identified by the Web Agent as explained on page 2.) and

- Data representing the static maximum allowed access for a given security descriptor and a corresponding client context in connection with an access request. (Page 2, Paragraph 3, under the title "user session cache " and Page 3. Paragraph 1-5) (Objects stored on local computers or network has security descriptor to help control access to the objects. Security descriptors include information about who owns the object, who can access it and in what way. On page 2, Paragraph 3, under the title "user session cache ", Netegrity discloses how the user is authenticated and begin access protected resources.)

Netegrity does not explicitly teach both the identifier and the security descriptor in resource access system.

However, in the same field of endeavor, Clifton discloses a resource access security system for controlling access to resources correspondingly assigned to address in an address spaces by the use of descriptors.(Column 3, lines 34-42; Abstract)

Furthermore Clifton discloses that the descriptor also includes information identifying an address space to which resources is assigned.(Column 3, lines 31-33) It would have been obvious to one having ordinary skill in the art, at the time the invention was made, to combine the features of the descriptors and identification as per teachings Clifton in to the method of as taught by Netegrity in order to secure the system.

16. As per claims 19, the combinations of Netegrity and Clifton discloses a method/a computer readable medium for enforcing static and dynamic access policy protecting a resource in a computer system as applied to claim 18 above. Furthermore, Netegrity discloses

Art Unit: 2132

the method wherein the storing of the static maximum allowed access data structure includes storing the static maximum allowed access data structure in cache memory. (Page 2)

## ***Conclusion***

**17. THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Samson B Lemma whose telephone number is 571-272-3806. The examiner can normally be reached on Monday-Friday (8:00 am---4: 30 pm). If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, BARRON JR GILBERTO can be reached on 571-272-3799. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR

Art Unit: 2132

system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

**SAMSON LEMMA**

*S.L.*

**06/09/2005**

*Gilberto Barron Jr.*

GILBERTO BARRON JR.  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100